

# Assignment by Malakounidis Paris

Github repo: <https://github.com/m3fisto/assignment>

The project was developed on a macbook Pro using the VScode IDE and was also tested on a Win10 pc.

The programming language is Python as requested in the exercise. Flask is used to create a simple API service and Pytest is used for both the functional and the non functional testing.

All the needed python dependencies can be found in requirements.txt located on the root folder of the project. Docker is also used for containerizing the service

## The service

A REST api server is created using Flask. It serves GET requests on 3 endpoints on <http://localhost:8800> The available endpoints are:

/people/<num>

/planets/<num>

/starships/<num>

Once the service is up it serves predefined json files

A server log file is created with the startup of the service on the logs folder of the project and is named server\_app.log. The log keeps track of HTTP requests on the server and the relevant response status code.

```
server_app.log
1 INFO:werkzeug: [31m [1mWARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead. [0m
2 * Running on all addresses (0.0.0.0)
3 * Running on http://127.0.0.1:8800
4 * Running on http://10.240.1.24:8800
5 INFO:werkzeug: [33mPress CTRL+C to quit [0m
6 INFO:werkzeug: * Restarting with stat
7 WARNING:werkzeug: * Debugger is active!
8 INFO:werkzeug: * Debugger PIN: 135-069-120
9 INFO:werkzeug:127.0.0.1 - - [12/Apr/2023 14:26:42] "GET /people/12/ HTTP/1.1" 200 -
10 INFO:werkzeug:127.0.0.1 - - [12/Apr/2023 14:33:29] "[33mGET /planets/200/ HTTP/1.1 [0m" 404 -
11
```

## Functional Tests

Functional tests were created in order to test happy and edge scenarios on the service. They are located in the /tests folder and named service\_test.py

A sample manual execution of the tests is the below(given the app is running):

```
pytest ./test/service_test.py --html=reportFT.html
```

and outputs:

```

parismalakounidis@C02CG1QAMD6R python_assignment % pytest ./test/service_test.py --html=reportFT.html
===== test session starts =====
platform darwin -- Python 3.9.16, pytest-7.1.2, pluggy-1.0.0
benchmark: 4.0.0 (defaults: timer=time.perf_counter disable_gc=False min_rounds=5 min_time=0.000005 max_time=1.0 calibration_precision=10 warmup=False warmup_iterations=100000)
rootdir: /Users/parismalakounidis/Desktop/projects/python_assignment, configfile: pytest.ini
plugins: reporter-html-0.8.2, playwright-0.3.0, html-3.2.0, base-url-2.0.0, docker-1.0.1, reporter-0.5.2, metadata-2.0.4, benchmark-4.0.0, flask-1.2.0
collected 7 items

test/service_test.py::test_get_people PASSED [ 14%]
test/service_test.py::test_get_people_error PASSED [ 28%]
test/service_test.py::test_get_planet PASSED [ 42%]
test/service_test.py::test_get_planet_error PASSED [ 57%]
test/service_test.py::test_get_starships PASSED [ 71%]
test/service_test.py::test_get_starships_error PASSED [ 85%]
test/service_test.py::test_wrong_format PASSED [100%]

----- generated html file: file:///Users/parismalakounidis/Desktop/projects/python_assignment/reportFT.html -----
===== 7 passed in 7.02s =====

```

An .html report is also created with filename reportFT.html:

#### reportFT.html

Report generated on 12-Apr-2023 at 14:54:02 by [pytest-html](#) v3.2.0

##### Environment

Base URL	
Packages	["pluggy", "1.0.0", "pytest", "7.1.2"]
Platform	macOS-13.2.1-x86_64-i386-64bit
Plugins	["base-url", "2.0.0", "benchmark", "4.0.0", "docker", "1.0.1", "flask", "1.2.0", "html", "3.2.0", "metadata", "2.0.4", "playwright", "0.3.0", "reporter", "0.5.2", "reporter-html", "3.0.8.2"]
Python	3.9.16

##### Summary

7 tests ran in 7.01 seconds.

(Un)check the boxes to filter the results.

☒ 7 passed ☐ 0 skipped ☐ 0 failed ☐ 0 errors ☐ 0 expected failures ☐ 0 unexpected passes

##### Results

Show all details / Hide all details

Result	Test	Duration	Links
<a href="#">Passed (show details)</a>	test/service_test.py::test_get_people	1.11	
<a href="#">Passed (show details)</a>	test/service_test.py::test_get_people_error	2.36	
<a href="#">Passed (show details)</a>	test/service_test.py::test_get_planet	0.13	
<a href="#">Passed (show details)</a>	test/service_test.py::test_get_planet_error	0.97	
<a href="#">Passed (show details)</a>	test/service_test.py::test_get_starships	2.08	
<a href="#">Passed (show details)</a>	test/service_test.py::test_get_starships_error	0.13	
<a href="#">Passed (show details)</a>	test/service_test.py::test_wrong_format	0.01	

## Performance Tests

Performance tests were created in order to measure the average response time of the server. They are located in the /tests folder and named stress\_test.py  
A random delay has been introduced in the server (between 0-2,5 secs per request)  
Sample manual execution of the tests is the following (given the service is up and running):

```

pytest ./test/stress_test.py --html=reportNFT.html --benchmark-min-rounds=50

```

and outputs:

```

parismalakounidis@C02CG1QAMD6R python_assignment % pytest ./test/stress_test.py --html=reportNFT.html --benchmark-min-rounds=50
===== test session starts =====
platform darwin -- Python 3.9.16, pytest-7.1.2, pluggy-1.0.0
benchmark: 4.0.0 (defaults: timer=time.perf_counter disable_gc=False min_rounds=50 min_time=0.000005 max_time=1.0 calibration_precision=10 warmup=False warmup_iterations=100000)
rootdir: /Users/parismalakounidis/Desktop/projects/python_assignment, configfile: pytest.ini
plugins: reporter-html-0.8.2, playwright-0.3.0, html-3.2.0, base-url-2.0.0, docker-1.0.1, reporter-0.5.2, metadata-2.0.4, benchmark-4.0.0, flask-1.2.0
collected 1 item

test/stress_test.py::test_api_sw PASSED [100%]

----- generated html file: file:///Users/parismalakounidis/Desktop/projects/python_assignment/reportNFT.html -----

benchmark: 1 tests
Name (time in ms)    Min      Max      Mean    StdDev   Median    IQR    Outliers  OPS      Rounds  Iterations
-----
test_api_sw         51.4464  2,507.8462  1,289.8075  691.8862  1,309.0098  1,078.7143  18;0  0.7753      50        1

Legend:
  Outliers: 1 Standard Deviation from Mean; 1.5 IQR (InterQuartile Range) from 1st Quartile and 3rd Quartile.
  OPS: Operations Per Second, computed as 1 / Mean

===== 1 passed in 68.16s (0:01:08) =====

```

On the above printout we can identify the Mean and standard deviation values.  
The test run for approximately one minute and the mean response time was 1.28 sec which is expected since the incurred delay on the server side is a random [0:2.5] seconds.

An .html report is also created with filename reportNFT.html:

## reportNFT.html

Report generated on 12-Apr-2023 at 14:59:50 by [pytest-html](#) v3.2.0

### Environment

Base URL	
Packages	{'pluggy': '1.0.0', 'pytest': '7.1.2'}
Platform	macOS-13.2.1-x86_64-i386-64bit
Plugins	{'base-url': '2.0.0', 'benchmark': '4.0.0', 'docker': '1.0.1', 'flask': '1.2.0', 'html': '3.2.0', 'metadata': '2.0.4', 'playwright': '0.3.0', 'reporter': '0.5.2', 'reporter-html': '0.8.2'}
Python	3.9.16

### Summary

1 tests ran in 68.15 seconds.

(Un)check the boxes to filter the results.

☒ 1 passed, ☐ 0 skipped, ☐ 0 failed, ☐ 0 errors, ☐ 0 expected failures, ☐ 0 unexpected passes

### Results

[Show all details](#) / [Hide all details](#)

Result	Test	Duration	Links
Passed <a href="#">(show details)</a>	test/stress_test.py::test_api_sw	67.98	

## Docker

A local docker image has to be created so the suite can run automatically. This can be done by executing on the root folder (where the dockerfile is located):

```
docker build -t api-sw .
```

This will create an image named api-sw. This image contains the API service

The image can be started manually via:

```
docker run -d --name api-sw -p 8800:8800 api-sw
```

This will create a container that listens on localhost:8800 and serves the app

## Test Suite execution

Given the prerequisite that the docker image has been created, we are able to run the tests with one command. This can be achieved with the pytest-docker plugin which spawns the container in order to run the tests and then terminates it in an approach similar to Ephemeral Environments. The log file from the container is fetched locally.

To run the complete Test Automation Suite (the app must not be running either manually or via docker) execute the following command:

```
pytest --html=report.html --benchmark-min-rounds=50
```

This will perform the following actions:

- Start the api-sw container
- Wait until it is reachable
- Run the functional tests
- Run the performance tests
- Stop and delete the container

and outputs:

```
parismalakounidis@C02CG1QAMD6R python_assignment % pytest --html=report.html --benchmark-min-rounds=50
===== test session starts =====
platform darwin -- Python 3.9.16, pytest-7.1.2, pluggy-1.0.0
benchmark: 4.0.0 (defaults: timer=time.perf_counter disable_gc=False min_rounds=50 min_time=0.000005 max_time=1.0 calibration_precision=10 warmup=False warmup_iterations=100000)
rootdir: /Users/parismalakounidis/Desktop/projects/python_assignment, configfile: pytest.ini
plugins: reporter-html1-0.8.2, playwright-0.3.0, html-3.2.0, base-url-2.0.0, docker-1.0.1, reporter-0.5.2, metadata-2.0.4, benchmark-4.0.0, flask-1.2.0
collected 9 items

Tests/fun_test.py::test_status_container PASSED [ 11%]
Tests/service_test.py::test_get_people PASSED [ 22%]
Tests/service_test.py::test_get_people_error PASSED [ 33%]
Tests/service_test.py::test_get_planet PASSED [ 44%]
Tests/service_test.py::test_get_planet_error PASSED [ 55%]
Tests/service_test.py::test_get_starships PASSED [ 66%]
Tests/service_test.py::test_get_starships_error PASSED [ 77%]
Tests/service_test.py::test_wrong_format PASSED [ 88%]
Tests/stress_test.py::test_api_sw PASSED [100%]

----- generated html file: file:///Users/parismalakounidis/Desktop/projects/python_assignment/report.html -----

----- benchmark: 1 tests -----
Name (time in ms)    Min      Max      Mean    StdDev   Median      IQR  Outliers  OPS  Rounds  Iterations
-----
test_api_sw         148.9654  2,449.6890  1,258.0766  687.2771  1,329.4593  1,230.0463  19;0  0.7949    50        1

Legend:
Outliers: 1 Standard Deviation from Mean; 1.5 IQR (InterQuartile Range) from 1st Quartile and 3rd Quartile.
OPS: Operations Per Second, computed as 1 / Mean

===== 9 passed in 84.58s (0:01:24) =====
```

## A combined html report was created with filename report.html:

### report.html

Report generated on 12-Apr-2023 at 16:13:10 by [pytest-html](#) v3.2.0

#### Environment

Base URL	
Packages	["pluggy": "1.0.0", "pytest": "7.1.2"]
Platform	macOS-13.2.1-x86_64-i386-64bit
Plugins	["base-url": "2.0.0", "benchmark": "4.0.0", "docker": "1.0.1", "flask": "1.2.0", "html": "3.2.0", "metadata": "2.0.4", "playwright": "0.3.0", "reporter": "0.5.2", "reporter-html1": "0.8.2"]
Python	3.9.16

#### Summary

9 tests ran in 84.53 seconds.

(Un)check the boxes to filter the results.

☒ 9 passed, ☐ 0 skipped, ☐ 0 failed, ☐ 0 errors, ☐ 0 expected failures, ☒ 0 unexpected passes

#### Results

[Show all details](#) / [Hide all details](#)

Result	Test	Duration
Passed <a href="#">(show details)</a>	Tests/fun_test.py::test_status_container	10.52
Passed <a href="#">(show details)</a>	Tests/service_test.py::test_get_people	0.57
Passed <a href="#">(show details)</a>	Tests/service_test.py::test_get_people_error	0.08
Passed <a href="#">(show details)</a>	Tests/service_test.py::test_get_planet	2.30
Passed <a href="#">(show details)</a>	Tests/service_test.py::test_get_planet_error	0.80
Passed <a href="#">(show details)</a>	Tests/service_test.py::test_get_starships	1.16
Passed <a href="#">(show details)</a>	Tests/service_test.py::test_get_starships_error	0.70
Passed <a href="#">(show details)</a>	Tests/service_test.py::test_wrong_format	0.01
Passed <a href="#">(show details)</a>	Tests/stress_test.py::test_api_sw	68.16